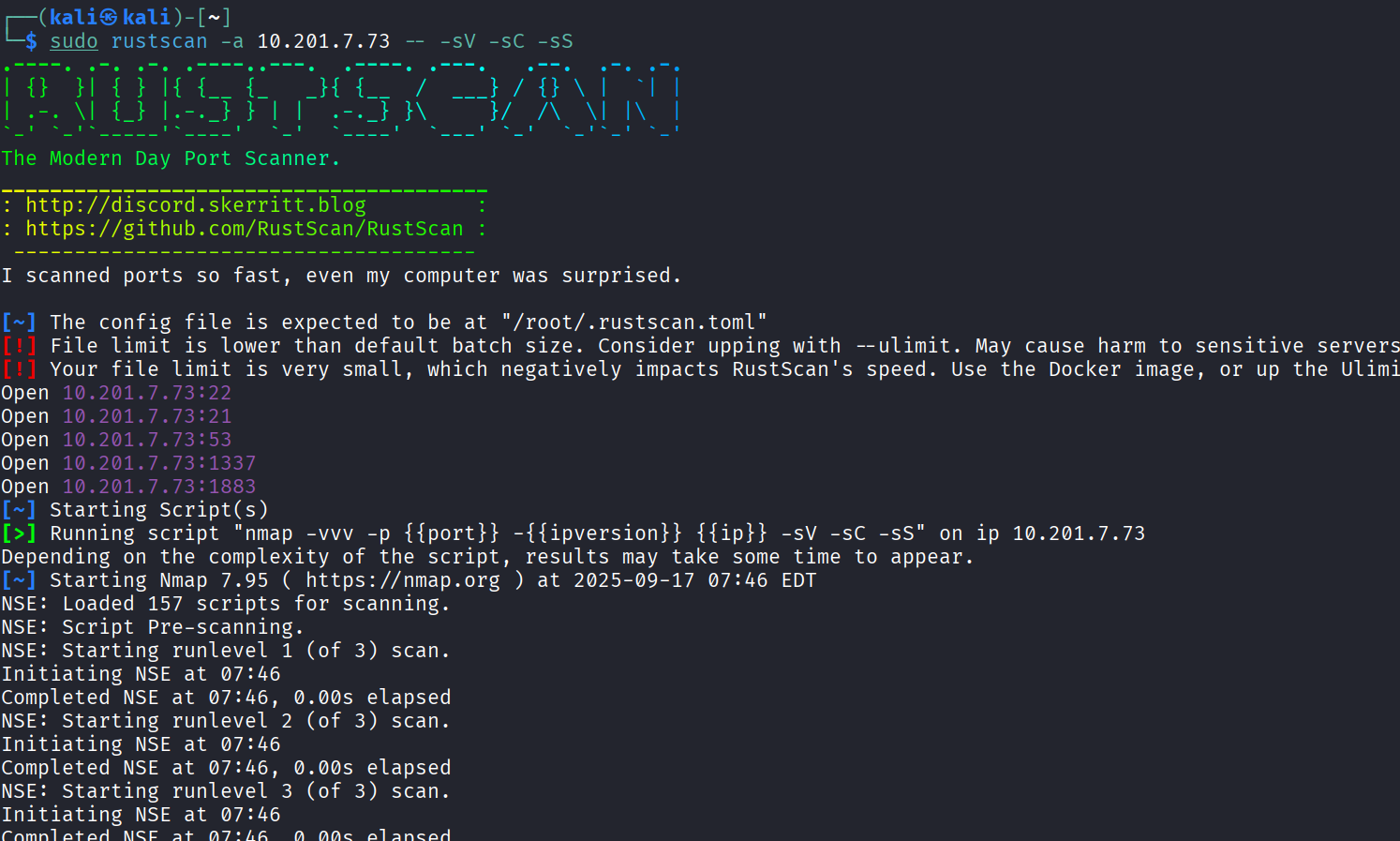
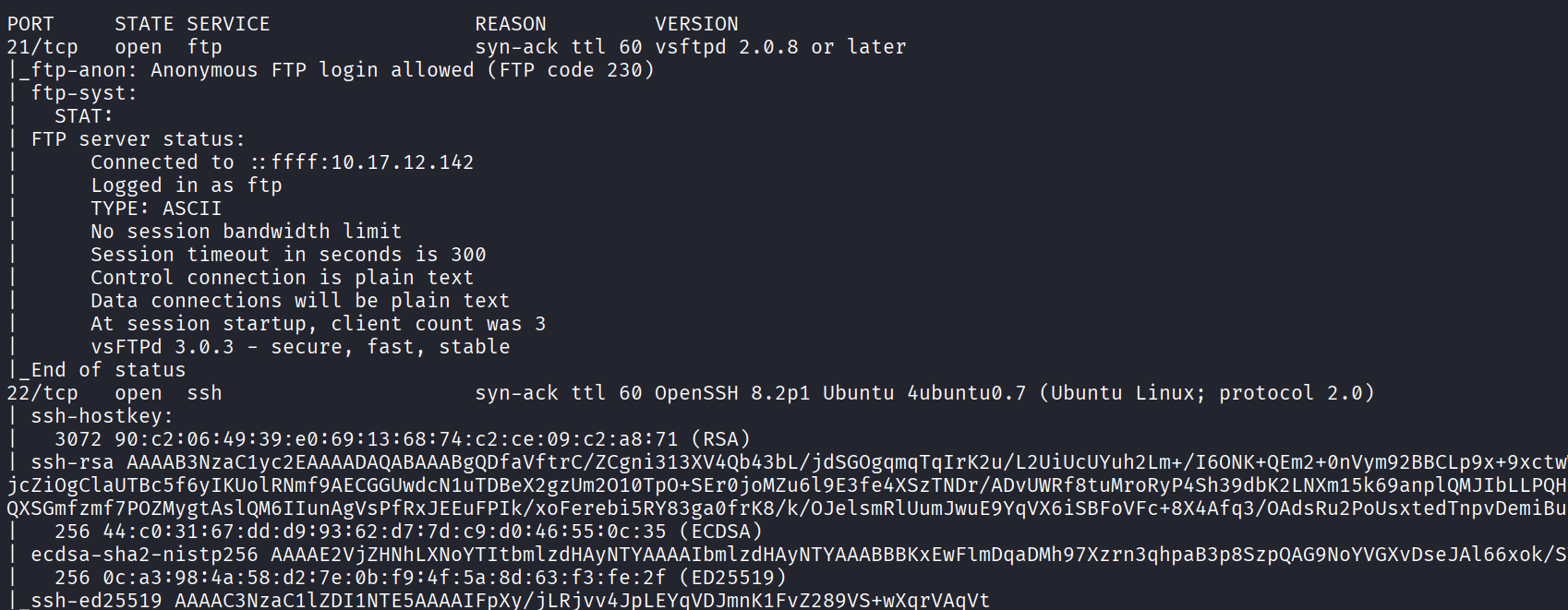
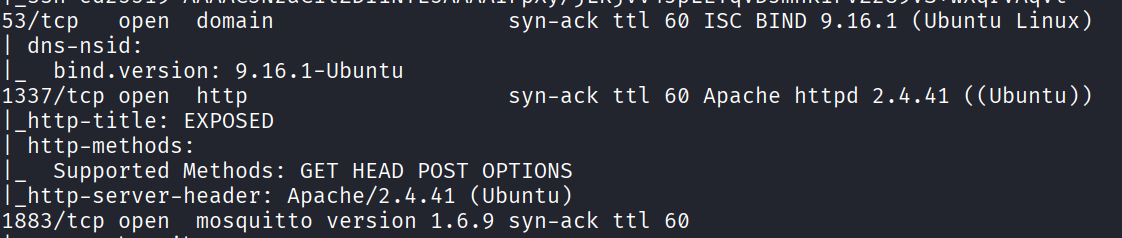
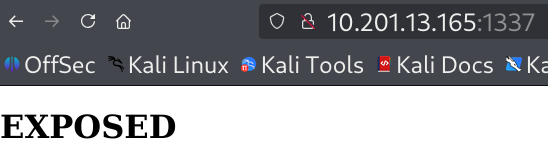
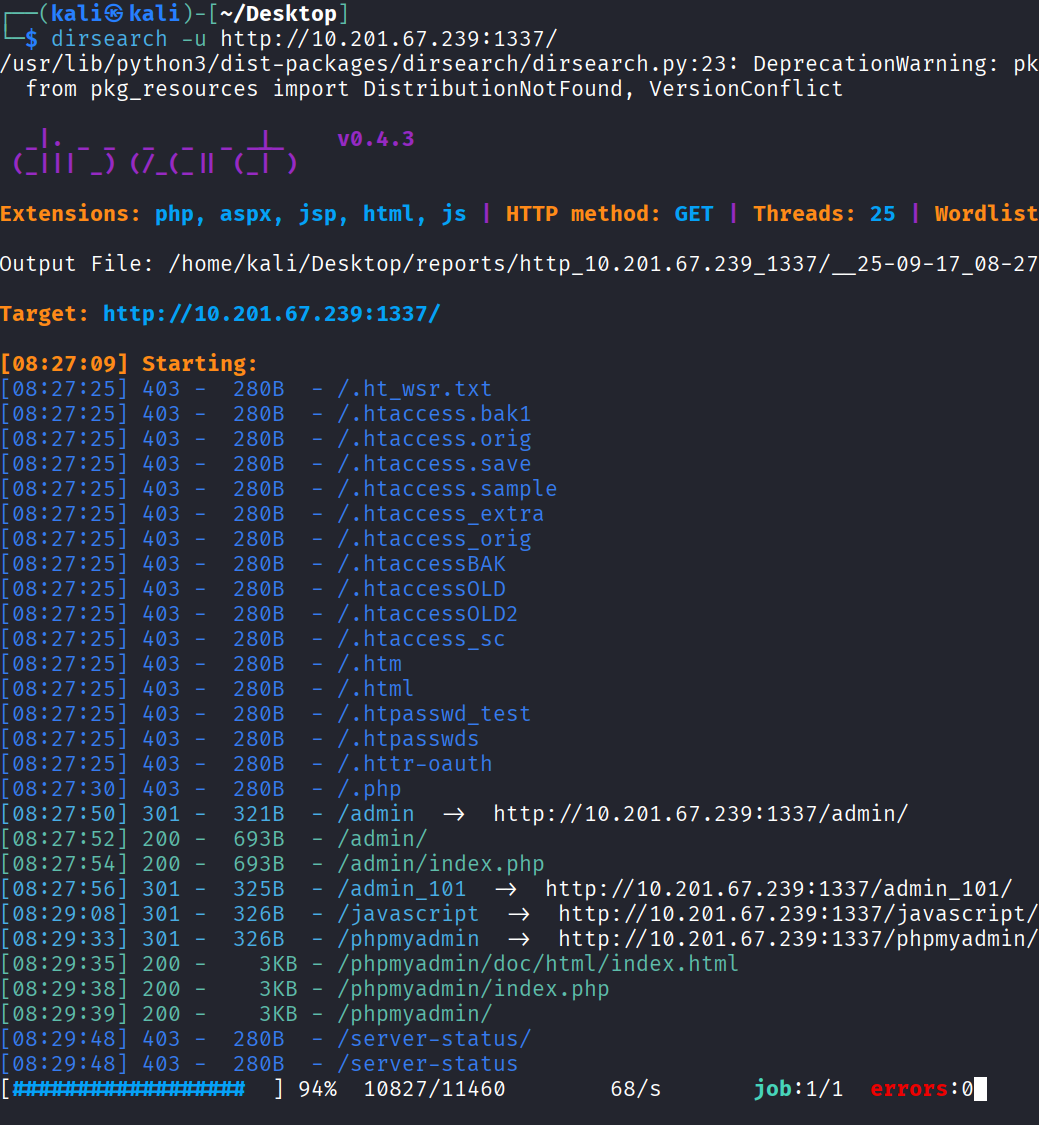
TryHackMe room: Expose  
Difficulty: Easy  
Room URL: <https://tryhackme.com/room/expose>  
  
Step 1: Active Recon - Port scanning  
  


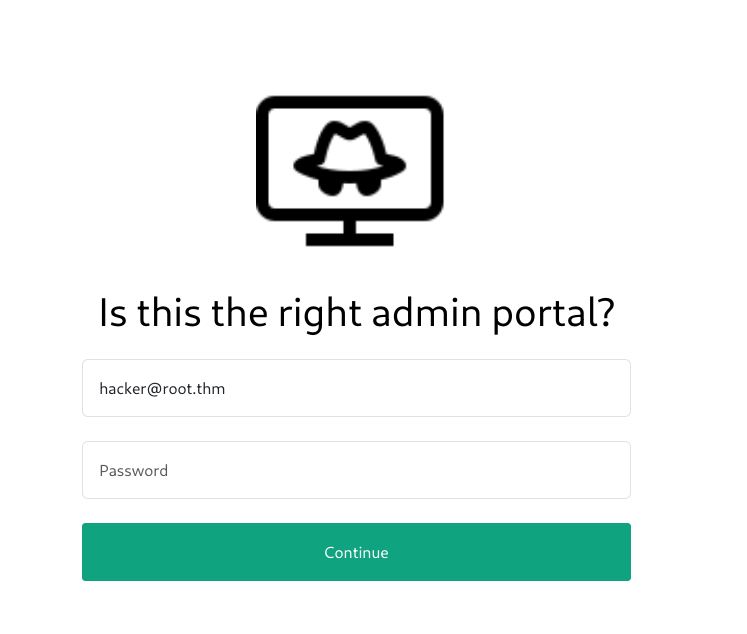




From the port scanning result, we can tell that http port is opened. Let’s access it on browser.   


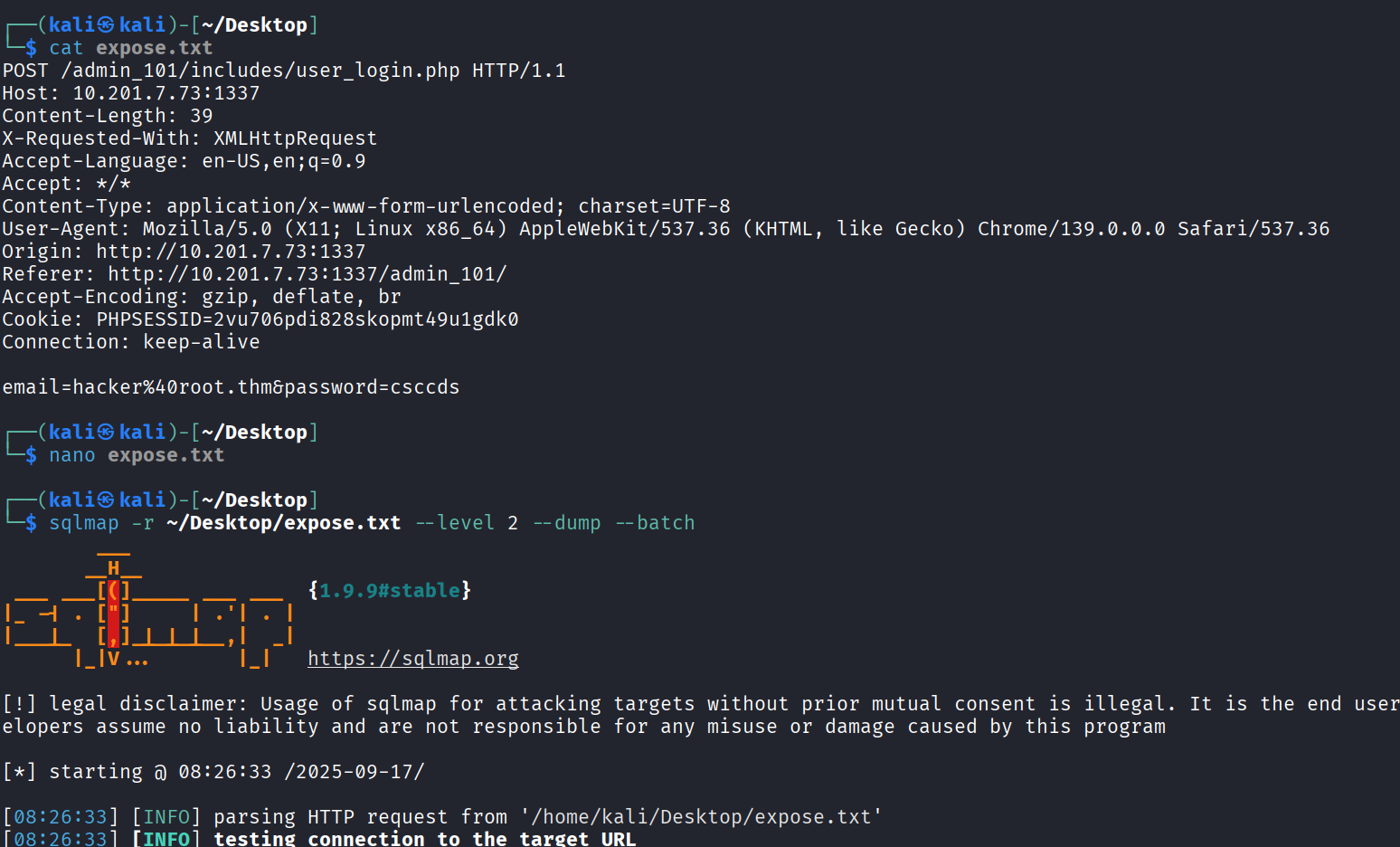
Step 2: enumerating directories  


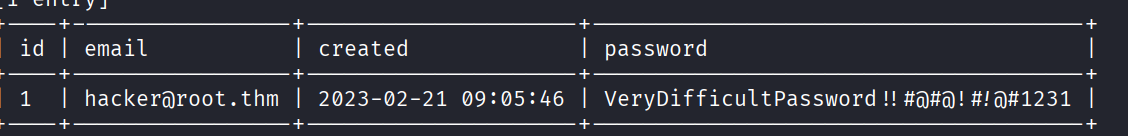
Let enumerate hidden directories from the url since we did not find anything useful on the web page. The path /admin\_101 stands out as interesting to me. Let’s navigate to it.

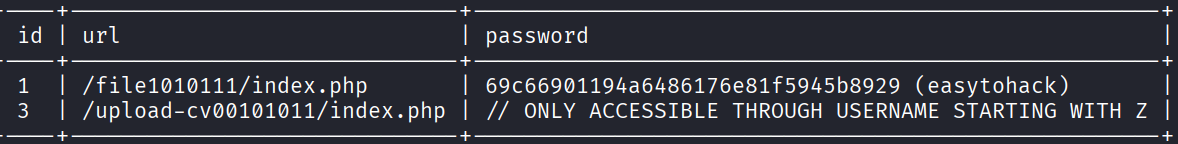


Step 3: Capture packet and response using Burp Suite  
  
Randomly key in any password and capture the packet using Burp Suite. From the response, we can tell that it may be vulnerable to sql injection attack. Let’s test it using sqlmap.   

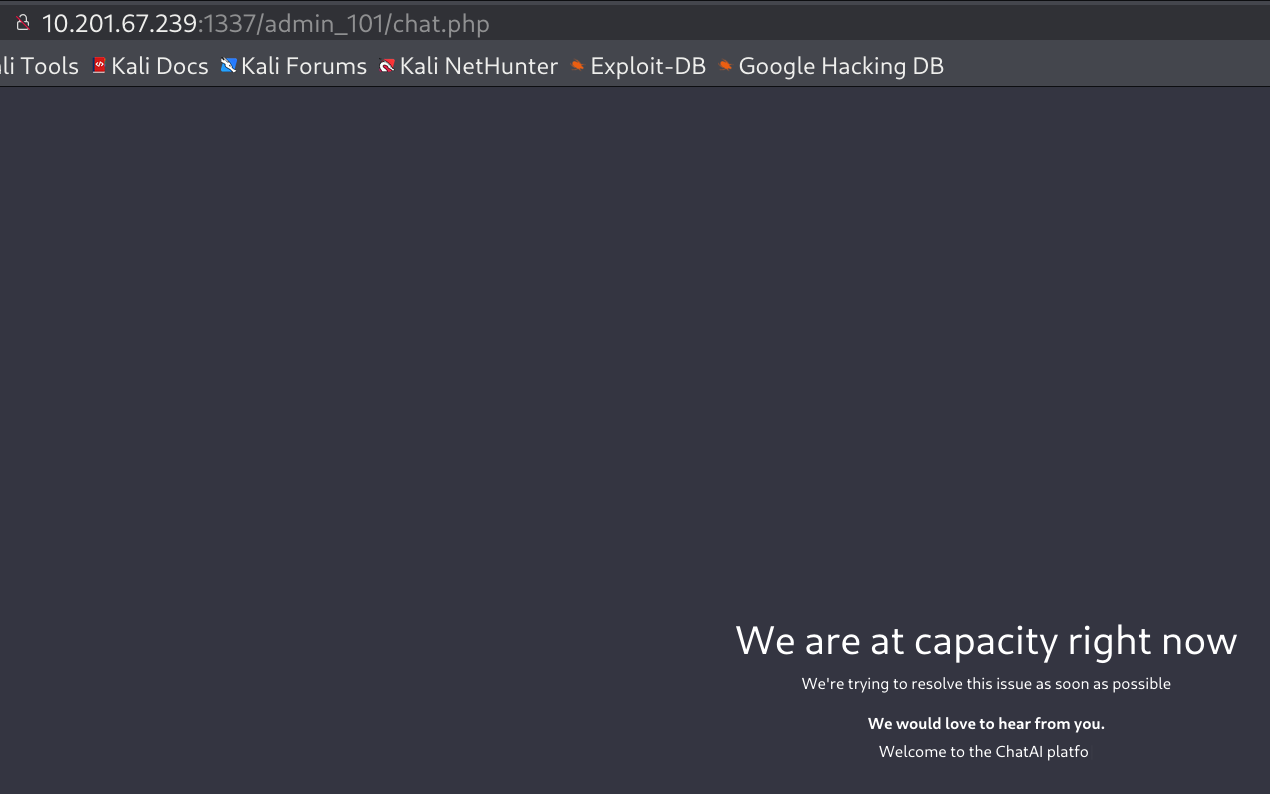

Step 4: Testing for SQL vulnerabilities using SQLMAP

Firstly, I copied and pasted the response in a file, expose.txt. Then I run the command, **sqlmap -r ~/Desktop/expose.txt –level 2 –dump –batch** for sql injection enumeration. From the result below, I have obtained the password for [hacker@root.thm](mailto:hacker@root.thm). Let’s login with the credentials now.   


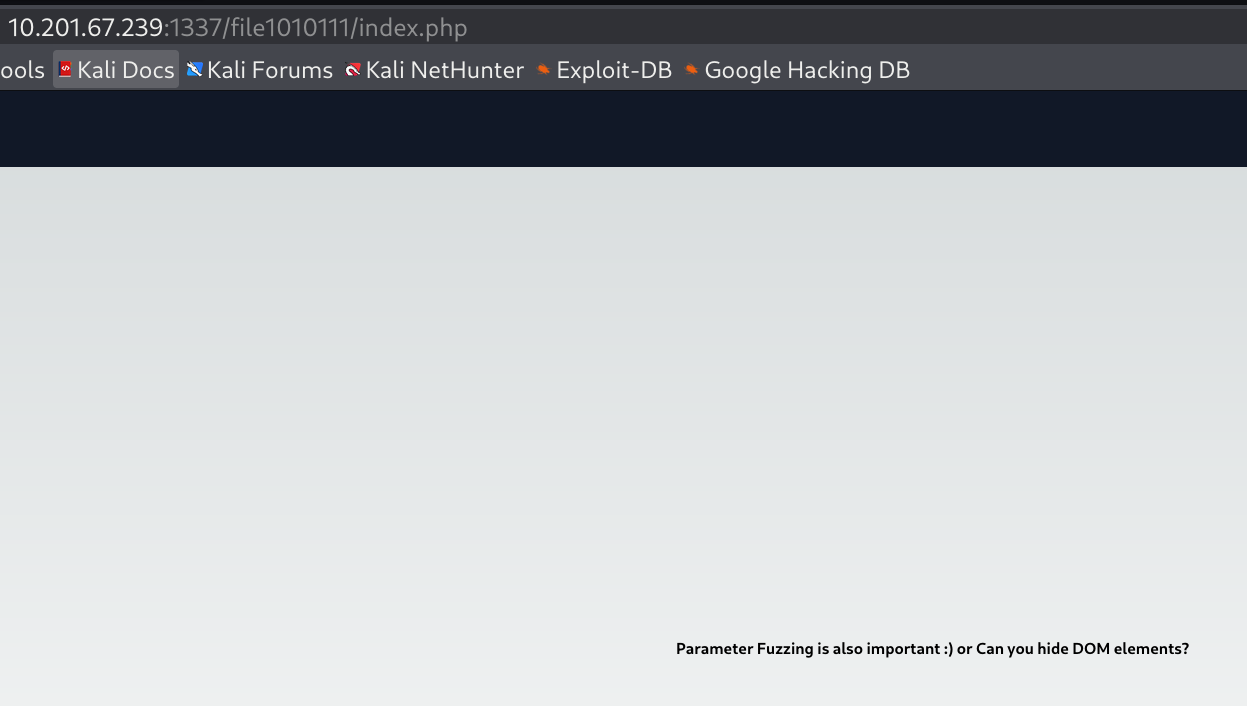


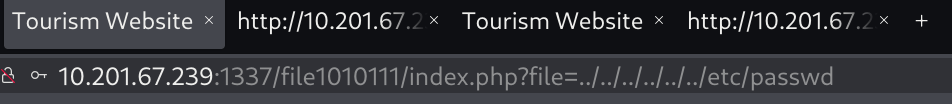


Step 5: enumerating the web page  
  
I have login the page and I did not discover any valuable information. Let’s proceed with the additional information from SQLMAP.



Let’s navigate to the path /file1010111 given by sqlmap. From the hint, we can tell that it is vulnerable to path traversal attack.

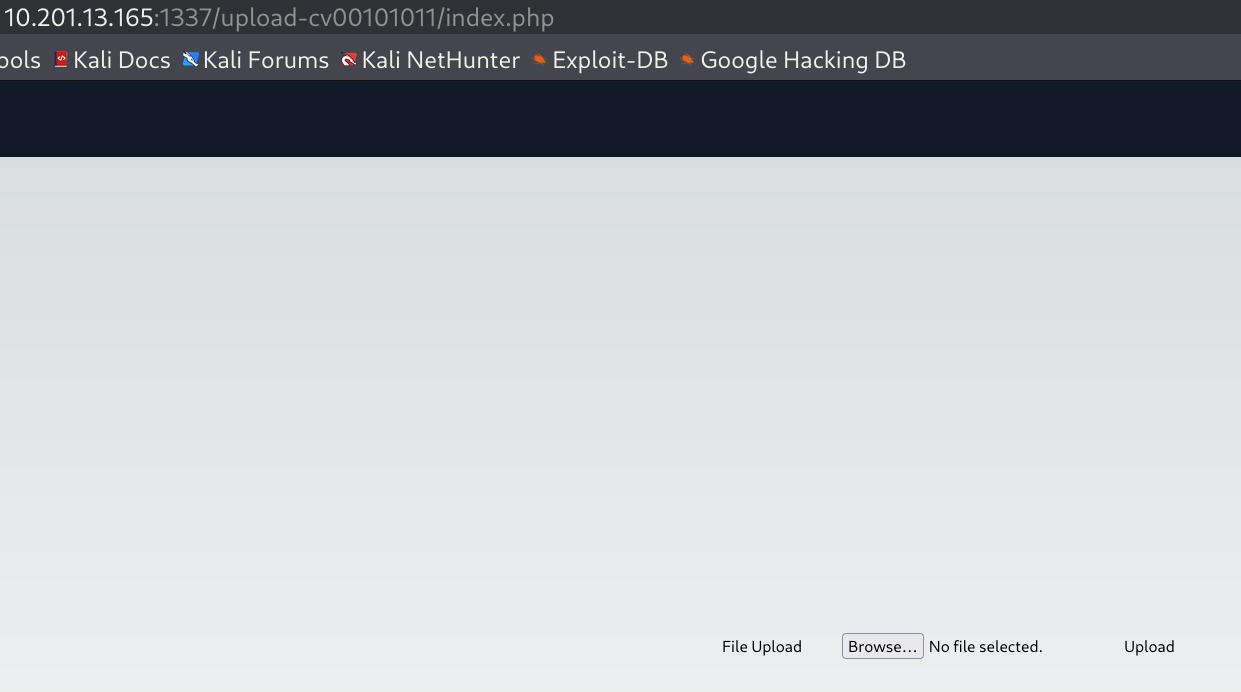
  

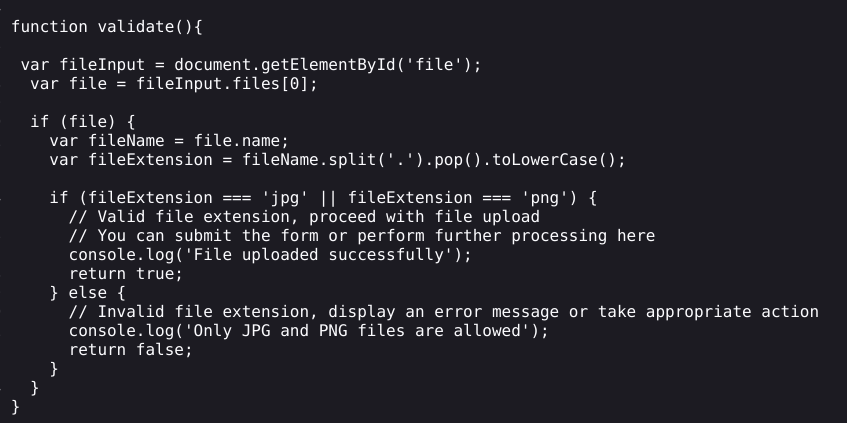



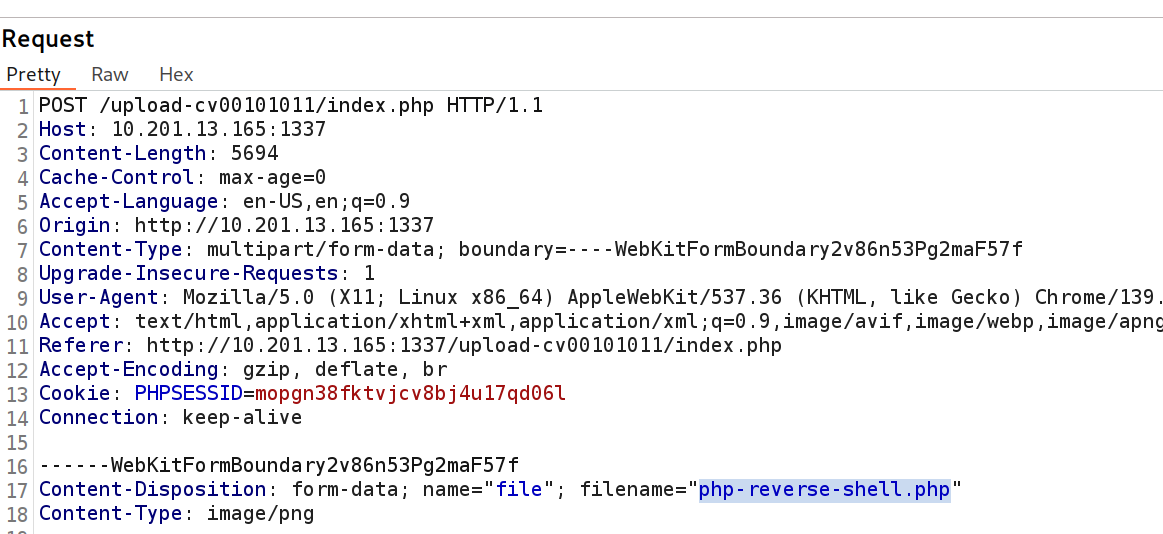
Now, I have revealed the username, zeamkish. We can use that as the password for the next path.

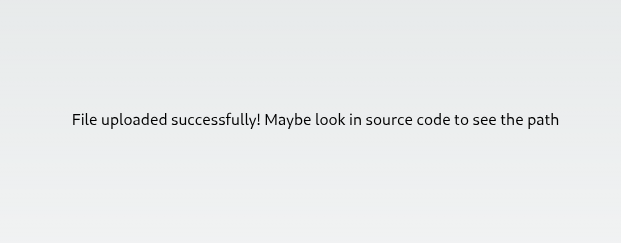


Navigate to the next path given by sqlmap, /upload-cv00101011 and enter the password, zeamkish. According to the source code, I can only upload png or jpg file extension. I renamed the file “php-reverse-shell.php” to “php-reverse-shell.php.png” and uploaded the file. I used burp suite to capture the file uploaded and modify the request file to php-reverse-shell.php

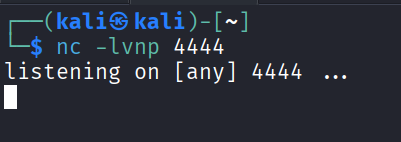


  
  
I captured the packet sent and modified the request to .php file. Upon successful upload, you will see the message shown in the screenshot below. Let’s view the source code now to discover the upload path.



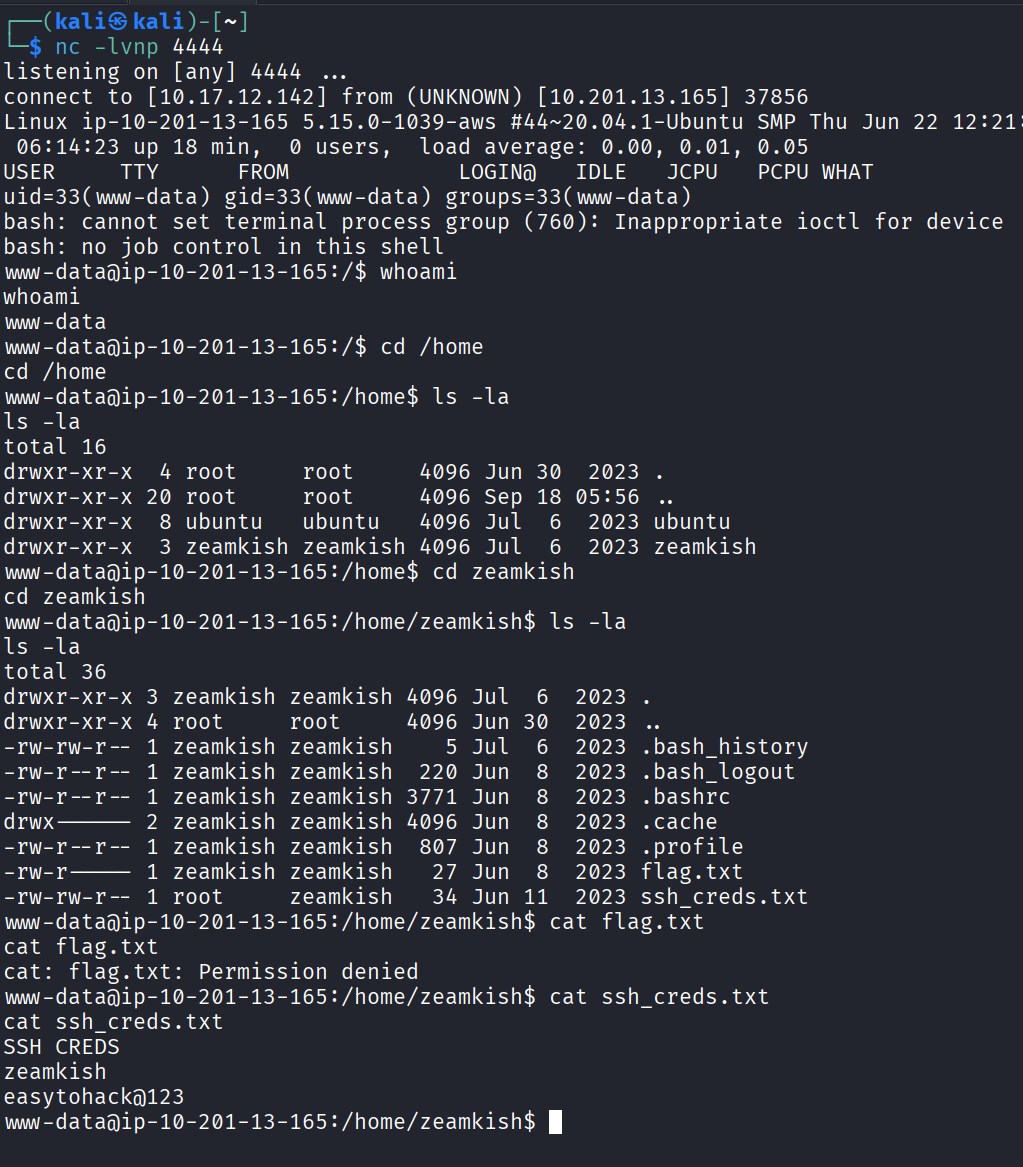


Step 6: Gaining user shell through reverse shell  
  
Before navigating to the path /upload\_thm\_1001, let’s start a listening port.

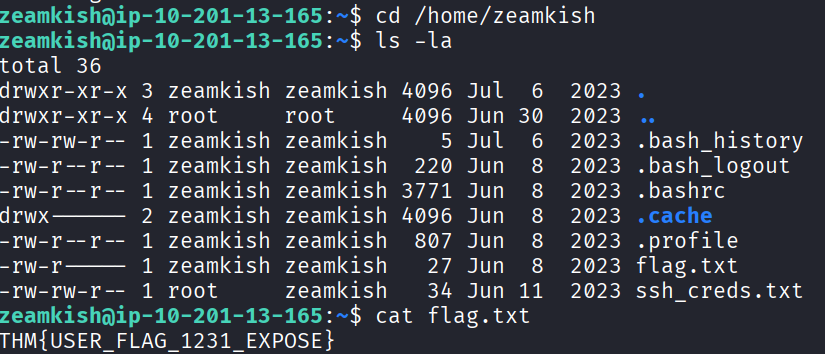


Click on the php-reverse-shell.php to execute the script.

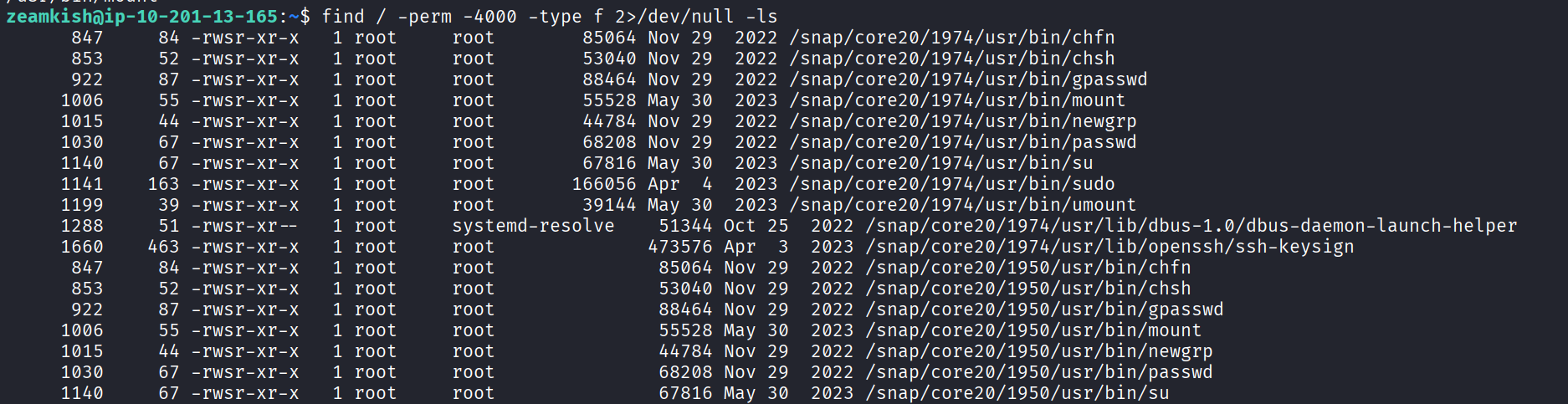
  
  
Now we have gain the user shell but it is not over yet. We need to gain zeamkish credentials.

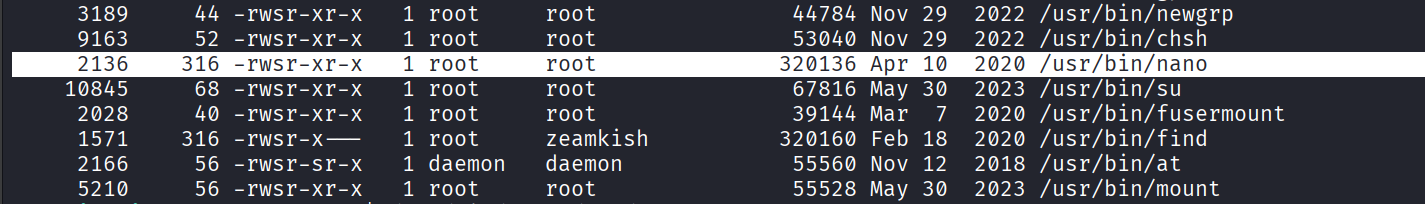


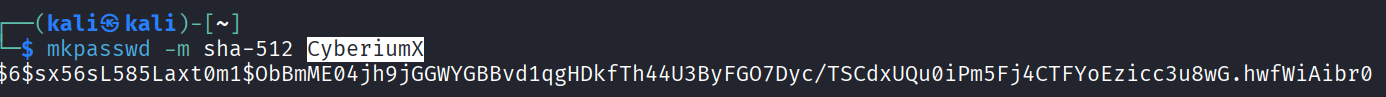
Now, let’s ssh into the zeamkish and get the user flag.

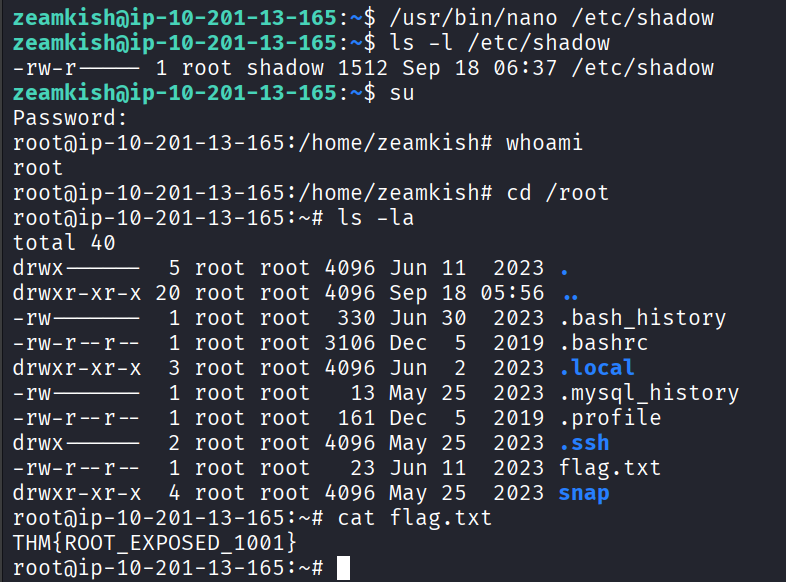


Step 7: Privilege Escalation via SUID  
  
Look for SUID set for binary files. Command: **find / -perm -4000 -type f 2>/dev/null**



  
  
I have found that nano has SUID which can be used to modify /etc/shadow to the hash of “CyberiumX” I will use mkpasswd to generate the hash.   
Command: **mkpasswd -m sha-512 CyberiumX**



  
  
  
Alternatively, we can also run the command: **/usr/bin/find . -exec /bin/sh -p \; -quit** to escalation privilege according to gtfo bin.   
